

Rangkuman PBO Pertemuan 1-6

Muhammad Nabil Omizar

2IA11

51422097

Pertemuan 1-2

Pemrograman berorientasi objek (Object Oriented Programming) adalah paradigma pemrograman yang berorientasikan kepada objek yang merupakan suatu metode dalam pembuatan program, dengan tujuan untuk menyelesaikan kompleksnya berbagai masalah program yang terus meningkat.

Pemrograman berorientasi objek ditemukan pada Tahun 1960, dimana berawal dari suatu pembuatan program yang terstruktur (structured programming). Metode ini dikembangkan dari bahasa C dan Pascal. Pemrograman berorientasi objek bekerja dengan baik ketika dibarengi dengan Objek-Oriented Analysis And Design Process (OOAD).

PBO:

- Menggabungkan fungsi dan data dalam kelas - kelas atau objek - objek
- Memiliki ciri Encapsulation (pengemasan), Inheritance (penurunan sifat) dan Polymorphism (perbedaan bentuk dan perilaku)
- Struktur program ringkas, cukup dengan membuat Objek dan class lalu bekerja berdasarkan object dan class tersebut
- Kode program sangat re-usable. object dan class dapat digunakan berkali-kali, sehingga dapat menghemat space memori.
- Efektif digunakan untuk menyelesaikan masalah besar, karena OOP terdiri dari class-class yang memisahkan setiap kode program menjadi kelompok - kelompok kecil, sesuai dengan fungsinya
- Sulit diawal (karena harus membuat class) namun selanjutnya akan terasa mudah dan cepat
- Eksekusi lebih cepat karena dieksekusi bersamaan, program hanya mengatur Objek, properties dan method-nya saja

Pemrograman Terstruktur:

- Memecah program dalam fungsi dan data
- Memiliki ciri Sequence (berurutan), Selection (pemilihan) dan Repetition (perulangan)
- Struktur program rumit karena berupa urutan proses dan fungsi-fungsi - Re-use kode program kurang
- Efektif digunakan untuk menyelesaikan masalah kecil dan tidak cocok untuk menyelesaikan masalah yang rumit, karena nantinya akan kesulitan menemukan solusi permasalahan ketika terjadi error
- Mudah diawal, namun kompleks diproses selanjutnya
- Eksekusi lebih lambat karena setiap perintah dikerjakan berurutan

Class

Class adalah cetak biru atau blueprint dari object. Class digunakan hanya untuk membuat kerangka dasar.

Sebagai analogi, class bisa diibaratkan dengan kendaraan. Kita tahu bahwa kendaraan memiliki ciri-ciri seperti merk, memiliki mesin, memiliki ban, dan beberapa ciri khas lain yang menyatakan sebuah benda tersebut adalah kendaraan. Selain memiliki ciri-ciri, sebuah kendaraan juga bisa dikenakan tindakan, seperti: menghidupkan mesin atau mematikan mesin.

Objek

Objek adalah entitas yang memiliki atribut, karakter dan kadang kala disertai kondisi. Objek mempresentasikan sesuai kenyataan seperti siswa, mempresentasikan dalam bentuk konsep seperti merek dagang, juga bisa menyatakan visualisasi seperti bentuk huruf.

Objek mirip dengan suatu rekaman (record) dalam suatu sistem berkas (contoh: rekaman karyawan). Suatu objek didefinisikan berdasarkan namanya atau menggunakan kata benda, objek juga memiliki atribut dan metode

Pada dasarnya ada dua karakteristik yang utama pada sebuah objek , yaitu:

Setiap objek memiliki atribut sebagai status yang kemudian akan disebut sebagai state.

Setiap objek memiliki tingkah laku yang kemudian akan disebut sebagai behaviour.

Pertemuan 3-6

Inheritance

Pewarisan merupakan sebuah bentuk “penggunaan kembali” (reusability); dimana class baru dibuat dari class yang pernah ada yang (biasanya) ditambah fasilitasnya.

Dalam pewarisan, constructor tidak diwariskan pada class turunannya, kecuali jika digunakan perintah super

Pewarisan tunggal (single inheritance) merupakan pewarisan dari satu class pokok (induk).

Pewarisan ganda (multiple inheritance) merupakan pewarisan dari dua atau lebih class pokok. Java tidak mendukung multiple inheritance.

Class yang mewariskan disebut dengan superclass / parent class / base class, sedangkan class yang mewarisi (class yang baru) disebut dengan subclass / child class / derived class.

Untuk menerapkan inheritance, gunakan statement “extends”. Keyword “super” digunakan oleh subclass untuk memanggil constructor, atribut dan method yang ada pada superclass-nya.

Encapsulation

Enkapsulasi merupakan cara untuk melindungi property (atribut) / method tertentu dari sebuah kelas agar tidak sembarangan diakses dan dimodifikasi oleh suatu bagian program.

Cara untuk melindungi data yaitu dengan menggunakan access modifiers (hak akses). Ada 4 hak akses yang tersedia, yaitu default, public, protected, private

Polymorphism

Overloading adalah diperbolehkannya dalam sebuah class memiliki lebih dari satu nama function/method yang sama tetapi memiliki parameter/argument yang berbeda.

Overriding method adalah kemampuan dari subclass untuk memodifikasi method dari superclass-nya, yaitu dengan cara menumpuk (mendefinisikan kembali) method superclass-nya.